Multi-stage optimal design of road networks for automated vehicles with elastic multi-class demand

Madadi, Bahman; van Nes, Rob; Snelder, Maaike; van Arem, Bart

DOI
10.1016/j.cor.2021.105483

Publication date
2021

Document Version
Final published version

Published in
Computers and Operations Research

Computers & Operations Research 136 (2021) 105483

Contents lists available at ScienceDirect

Computers and Operations Research

journal homepage: www.elsevier.com/locate/cor

Multi-stage optimal design of road networks for automated vehicles with elastic multi-class demand

Bahman Madadi [a,*], Rob van Nes [a], Maaike Snelder [a,b], Bart van Arem [a]

[a] Delft University of Technology, Civil Engineering Faculty, Transport & Planning Department, 2628 CN Delft, the Netherlands
[b] Netherlands Organization for Applied Scientific Research (TNO), 2628 CK Delft, the Netherlands

ARTICLE INFO

Keywords:
Time-dependent network design problem
Automated vehicles
Mixed traffic
Transportation
Evolutionary computations

ABSTRACT

With the advent of automated vehicles (AVs), new infrastructure planning concepts such as subnetworks of AV-ready roads have been proposed to improve the performance of transportation networks and to promote the adoption of AVs. However, these subnetworks should evolve over time in response to the growing AV demand, which necessitates a multi-stage modeling approach. In this study, we propose multi-stage deployment of AV-ready subnetworks and formulate it as a time-dependent network design problem, which is a bi-level mixed-integer programming problem. The lower level is a simultaneous travel mode and route choice equilibrium with continuous decision variables, and the upper level is a design problem including infrastructure investment decisions to determine which roads to upgrade and include in AV-ready subnetworks for mixed traffic. We use a case study of a real road network to demonstrate the concept. Since computational efficiency is a key factor for solving such large-scale problems, we develop two efficient and tailored evolutionary heuristics to solve the problem, and compare their performance to a computationally demanding Genetic-algorithm-based solution method. The results indicate that the proposed algorithms can efficiently solve this large-scale problem while satisfying constraints in all scenarios, and outperform Genetic algorithm, particularly in the scenario with larger number of stages. Moreover, in all scenarios, deployment of AV-ready subnetworks leads to improvements in network performance in terms of total travel time and cost. However, the improvements are always accompanied with increased total travel distance. The extent of changes depends on AV market penetration rate, AV-ready subnetwork density and timing of densification.

1. Introduction

Automated vehicles (AVs) are on the horizon; however, it might take a long time before a large market share for highly automated vehicles can be observed. In the meantime, a heterogeneous mix of traffic with AVs and conventional vehicles (CVs) on the roads is inevitable. According to SAE International (2018), there are five levels of vehicle automation where level 5 denotes fully automated vehicles, levels 3–4 denote conditionally and highly automated vehicles, and levels 1–2 denote partially automated vehicles, which are already available on the market. Although development of flawless level-5 AVs can take a long time, levels 3–4 might become a reality within the coming decade (Shladover, 2016). Based on SAE International (2018), the operating design domain (ODD) of levels 3–4 is limited. However, there is currently a lack of data to specify the exact ODD limitations of AVs. On the other hand, while ODDs are the accepted language of the automotive industry to define functional requirements for vehicle automation, there is no universally accepted standard for road operators describing the readiness of road network infrastructure to support automation functions. Furthermore, the interactions between AVs and infrastructure become crucial during the long transition period to full automation, since a mix of CVs and AVs is then expected on the roads. A proper infrastructure can support AVs' functionality, extend their ODD and improve safety for all road users, while lack of proper infrastructure can have negative impacts on these factors.

Some studies have specified infrastructure requirements for safe operation of AVs (see (Farah et al. (2018) for a review). However, establishing a correspondence between the requirements and different parts of road networks can be challenging. Moreover, these requirements can be idealistic, expensive and difficult to meet, especially in some road types, such as local distributors. Other researchers have suggested optimal networks of dedicated lanes (Chen et al., 2016),

* Corresponding author.
E-mail addresses: b.madadi@tudelft.nl (B. Madadi), r.vannes@tudelft.nl (R. van Nes), m.snelder@tudelft.nl (M. Snelder), b.vanarem@tudelft.nl (B. van Arem).

https://doi.org/10.1016/j.cor.2021.105483
Received 13 August 2020; Received in revised form 6 June 2021; Accepted 13 July 2021
Available online 21 July 2021

dedicated links (Ye and Wang, 2018), and dedicated zones for AVs (Chen et al., 2017) as well as AV-ready subnetworks for mixed traffic (Madadi et al., 2020) to address this issue and promote the adoption of AVs via network design concepts. These are promising approaches; nonetheless, models representing these designs require more details and extensions to become operational. Furthermore, so far they have only been tested on small, often theoretical networks. Application of these concepts on large-scale real networks is a crucial but missing next step, especially since there are many practical issues and considerations involved with real networks that are not observed with theoretical networks, including network hierarchy and road type. Besides, dedicating parts of a network to one class of vehicles, can compromise accessibility of other classes and modes.

Moreover, designing optimal road networks for AVs is a gradual process that depends on the level of demand for AVs. Since the demand for AVs will increase over time, optimal networks for AVs should evolve over time as well. An efficient design for a network with a low level of AV demand is not necessarily an efficient design for the same network with a very high level of AV demand. In addition, demand for cars and the performance of road networks cannot be assessed without considering alternative modes.

On the other hand, analyses of automated driving system (ADS) disengagements and AV accident reports in the USA provide some insight into the suitability of various parts of road networks for AVs. Approximately, 10% of ADS disengagements can directly be associated with road infrastructure (Dixit et al., 2016), and around 56% have been due to system failures, which can also be attributed to vehicle's inability to cope with its surrounding. Furthermore, around 87% of all disengagements have occurred at interstate roads and urban streets, while motorways, freeways and arterial roads combined are associated with less than 13% of all disengagements (Favarò et al., 2018). According to Xu et al., (2019), 71.2% of the accidents involving AVs in California so far have occurred at intersections, 20.5% in urban streets and only 1.4% in highways. Moreover, based on the evidence of existing level-2 AVs, drivers are very likely to use the automated mode in freeways and unlikely to use it on rural and urban roads (Hardman et al., 2019). This shows that road type is an essential factor for safety of AVs.

In light of the discussion above, in this study, we select a set of roads based on their characteristics to define a (potentially) safe feasible region for the operation of AVs in automated driving mode (using ADS) in mixed traffic (on the same lanes as CVs). This region will be upgraded with physical and digital infrastructure investments to guarantee safety for all road users and to improve the efficiency of the automated driving mode. However, investing in all feasible roads can be costly and unnecessary. Therefore, a next step is required to make a selection among the feasible roads that optimizes the trade-off between the investments and the societal benefits they provide in order to construct an efficient AV-ready subnetwork for mixed traffic. Therefore, we propose a multi-stage mathematical model to optimize the evolution of AV-ready subnetworks over time in response to the gradual development of AV demand considering the competing transport modes. Moreover, we use a case study of a large-scale real road network to demonstrate our proposed design concept and to discuss practical considerations related to its deployment. This article contributes to the existing literature and extends the authors' previous works on AV-ready subnetworks (Madadi et al., 2021, 2020, 2019) by the following.

1. Proposing the multi-mode multi-class formulation of the network equilibrium problem with a mix of AVs and CVs including asymmetric link costs (lower level problem)
2. Proposing a multi-stage (time-dependent) model for optimizing AV-ready subnetworks for mixed traffic over time with endogenous and time-varying demand for AVs
3. Proposing two heuristic algorithms to solve the problem, comparing their performance to a Genetic-algorithm-based solution procedure, and providing a rigorous analysis of their performance using extensive computational experiments
4. Demonstrating the applicability of the proposed methodology on a large-scale case study of the real network of the Amsterdam metropolitan region

The rest of the manuscript is organized as follows: Section 2 includes a brief problem background, Section 3 presents the problem formulation and the solution methods, Section 4 demonstrates the case study and numerical results, and Section 5 entails the discussion and concluding remarks.

2. Problem background

The problem under consideration is designing optimal AV-ready subnetworks in road networks and their evolution over a planning horizon considering a time-varying demand from different modes and vehicle types. In previous studies, these subnetworks have been referred to as automated driving (AD) subnetworks as well. Since models used to address such problems include many components, we provide an overview of the relevant studies in this section to place the problem within the literature, and discuss additional studies pertinent to each model component in the corresponding section.

In transport literature, strategic decisions regarding transport networks are considered within the framework of the well-known network design problem (NDP), which involves a large body of literature. For review studies, the reader is referred to (Farahani et al., 2013; Magnanti and Wong, 1984; Yang and Bell, 1998). NDPs are generally modeled as Stackelberg leader–follower games where the leader tries to make optimal decisions for transport infrastructure considering the followers' (travelers) response to the changes in the network by their travel choices. Mathematically, this results in a bi-level non-convex NP-Hard problem, which is very challenging to solve (Yang and Bell, 1998). The upper level includes optimal decisions for transport networks (e.g., building new streets and adding new lanes) considering their cost (investment) as well as their benefits (e.g., total user travel time saving), and the lower level includes predicting the travelers' response to these decisions via their travel choices, which is commonly perceived to follow Wardrop's equilibrium principles (Wardrop, 1952).

Accordingly, NDPs can be classified into several variants. Based on the upper level decision variables, discrete NDP (DNDP) (Chen and Alfa, 1991; Leblanc, 1975; Miandoabchi et al., 2013), continuous NDP (CNDP) (Davis, 1994; Wang et al., 2014) and mixed NDP (MNDP) (Cantarella et al., 2006; López-Ramos et al., 2019) are recognized. Based on the upper level objective function, single objective (Cantarella et al., 2006) and multi-objective (Miandoabchi et al., 2013; Miandoabchi et al., 2012; Wang and Szeto, 2017) variants have been studied. Based on the lower lever equilibrium type, deterministic user equilibrium (DUE) (Leblanc, 1975; Li et al., 2018), stochastic user equilibrium (SUE) (Davis, 1994), system optimal (Dantzig et al., 1979) and mixed (Chen et al., 2017) versions have been proposed. Regarding the lower level demand, fixed demand (Tobin and Friesz, 1988) and elastic demand (Yang, 1997), single class (Chen and Alfa, 1991) and multi-class (Chen et al., 2017), as well as unimodal (Chen et al., 2016) and multi-modal (Miandoabchi et al., 2012) versions have been considered. Another distinction is based on the number of decision stages (time periods) which leads to single stage (most NDPs) and multi-stage or time-dependent NDP (NDP-T) (Lo and Szeto, 2009; Ukkusuri and Patil, 2009). Finally, in recent years, a number of NDPs have considered special network configurations such as dedicated lanes (Chen et al., 2016), dedicated links (Ye and Wang, 2018), and dedicated zones for AVs (Chen et al., 2017) as well as AV-ready subnetworks for mixed traffic (Madadi et al., 2020). This last category of NDPs is of special interest in this article, since they are closely related to this study due to proposing design concepts for AD. We refer to them as AD-NDP and briefly describe them in the following paragraph.

Chen et al. (2016) studied the problem of optimal deployment of dedicated AV lanes over time. The upper level included deciding where and when to deploy dedicated AV lanes to minimize the social cost and the lower level involved a multi-class DUE traffic assignment with fixed demand and a single mode (i.e., car). The model was tested on the (simplified version of) South Florida network with 232 links. To the best of our knowledge, this is the only time-dependent AD-NDP (AD-NDP-T) study so far. Chen et al. (2017) proposed optimizing dedicated AV zones in road networks with the upper level objective of minimizing social cost and a mixed route choice model for the lower level including system optimal routing for AVs and a deterministic routing for CVs. The model was demonstrated on a number of small synthetic networks (maximum 288 links). Ye and Wang (2018) studied the problem of optimizing dedicated AV links combined with congestion pricing for CV links where the upper level entailed minimizing total travel time cost as well as the link-based toll, and the lower level was a DUE including CVs and AVs. This model was tested on a small synthetic network with 18 links. It is worth noting that all three aforementioned configurations restrict CVs' access to some parts of the network since they dedicate some lanes, links and zones only to AVs. Finally, Madadi et al. (2020) proposed optimal AV-ready subnetworks referred to as "AD subnetworks", which were accessible for all vehicles, yet adjusted for optimal automated driving performance in mixed traffic (i.e., CVs and AVs on the same lanes). The upper level objective was to minimize total travel cost along with infrastructure adjustment cost of creating the subnetwork and the decision variables represented the links to be selected for the subnetwork. The lower level was a multi-class SUE with the path-size logit. This model was demonstrated on a semi-real network of Delft with 1151 links.

In this study, we model the problem as a multi-stage (time-dependent) discrete NDP (AD-DNDP-T) with multi-mode and multi-class demand involving AVs. To the best of our knowledge, this is the first NDP-T with multi-mode multi-class demand with asymmetric link costs (for CVs and AVs) involving special network configurations for AVs with demonstrations on a large-scale real road network.

The AD-DNDP-T problem considered in this study is a multi-stage mixed-integer bi-level network design problem where both levels individually are non-convex, the upper level decision variables appear in the lower level, and the upper level includes a non-differentiable connectivity constraint.

In general, bi-level problems are very difficult to solve for optimality. Therefore, heuristic solutions have been commonly used to solve these problems in the past (Migdalas, 1995). Recent developments in bi-level programing have made it possible to solve reasonable-sized bi-level problems with linear objectives and constraints for optimality (Fischetti et al., 2018a, 2017), and new methods have been developed to speed up the search process in mixed-integer programming problems (Fischetti et al., 2018b). However, the linearity condition is not met in our problem. Even the high-point relaxation method for finding lower bounds cannot be applied here since it requires convexity of lower and upper level objectives and constraints individually. Moreover, such methods cannot cope with the connectivity constraint of the problem considered in this study. For a systematic review of bi-level problems, the reader is referred to Lachhwani and Dwivedi (2018).

For small instances of single-stage DNDPs, Wang et al. (2013) developed two global optimization methods based on system optimal and user equilibrium relaxations. For large-scale single-stage DNDPs, heuristics based on the genetic algorithm (GA) and simulated annealing (SA) have been the most common solutions so far (Farahani et al., 2013).

DNDP-Ts on the other hand, have rarely been studied in the literature and are among the most challenging NDPs to solve. Therefore, in DNDP-T studies so far, either small case studies have been used that can be solved manually or approximate solutions have been used to solve the problem. Szeto et al., (2010) have used PREMIUM SOLVER PLATFORM to solve a DNDP-T for a small synthetic network with four links. Miandoabchi et al., (2015) have used two metaheuristics, namely, non-dominated sorting GA and a B-cell algorithm on seven test networks with maximum 66 links to solve a multi-objective DNDP-T. O'brien and Yuen (2007) have used a combination of branch and bound, and generalized reduced gradient for solving a DNDP-T. Finally, Chen et al. (2016) have solved a DNDP-T for the network of South Florida with 232 links using the active-set algorithm. It is crucial to notice that for combinatorial optimization problems such as the one considered here, computation times increase exponentially with respect to the problem size (i.e., number of integer decision variables). Therefore, for solving case studies with large-scale real networks, efficiency of the solution method is of paramount importance. On the other hand, design of AV-ready subnetworks imposes subnetwork connectivity constraints (Madadi et al., 2020), which make most existing solutions ineffective since they are not designed to generate connected graphs. A generic remedy is using penalty functions to penalize disconnected solutions, but this can affect the efficiency of the solution method.

Therefore, in this study, we solve the problem using three evolutionary algorithms that are designed to deal with the complexity of the problem and the requirements mentioned earlier. The methodology and the solution algorithms are described in the following section.

3. Multi-stage design of subnetworks for automated driving

The concept of AV-ready subnetworks or AD subnetworks for mixed traffic entails selecting a subset of (potentially safe) roads within road networks to form a subnetwork, and upgrading this subnetwork with necessary (physical and digital) infrastructure adjustments to meet higher quality standards to ensure uninterrupted, safe and efficient AD using ADS in mixed traffic conditions. The extent of these adjustments depends on available funds and authorities' commitment to facilitating safe AD. Several lists of possible adjustments under various scenarios based on expert interviews are provided by Lu et al. (2019). In this study, the selection of roads for adjustment occurs in two steps. The first step includes a preselection of feasible road segments (links) based on road type to guarantee that all selected links for the subnetwork have the potential to meet the desired standards after reasonable adjustments. It entails excluding roads with complex interactions between AVs and other road users. A more elaborate discussion on road selection for AV-ready subnetworks can be found in Madadi et al. (2019) and a practical example is shown here in Section 4 for a case study of Amsterdam. Since infrastructure adjustment projects tend to be costly and time consuming, a second step is required to select the best combination and timing of the adjustments that maximize their total societal benefits. The following subsections include the mathematical formulation of this problem. Table 1 provides the full notation.

3.1. General assumptions

Let $G(N,A)$ denote a strongly connected graph representing the transportation network where N is the set of nodes and A is the set of directed arcs (links) representing the road and transit line segments. The planning horizon is divided into T equal decision stages (time periods) each having a length of l years. For each stage τ, a decision is made to select a subset of roads denoted by A_1^τ for the AV-ready subnetwork represented by the graph $G_1^\tau(N_1^\tau, A_1^\tau)$ and the rest of the links are denoted by$A_0^\tau(A = \{A_0^\tau \cup A_1^\tau\}, A_0^\tau \cap A_1^\tau = \varnothing)$. An example of an AV-ready subnetwork graph is shown in Fig. 1. The effects of the construction period, which can be short given that the adjustments are not major, are ignored.

All vehicles are allowed on all links in the network but AVs can switch to automated driving mode (i.e., activate their ADS) only on the AV-ready subnetwork links, which are adjusted for safe and efficient AD in mixed traffic. This means that on AV-ready subnetwork links, there will be a mix of CVs driving manually and AVs using their ADS. Moreover, on the rest of the links, both CVs and AVs drive manually.

Table 1
Notation.

Notation	Definition
Sets	
W	Set of origin–destination (OD) pairs w
$R_m^{\tau,w}$	Set of routes r between OD pair w for mode m in stage τ
M	Set of modes $m \in \{0,1,2\}$ (0 = PT, 1 = CD, 2 = AD)
K	Set of user classes $k \in \{0,1,2\}$ (0 = no access to vehicle, 1 = access to CV, 2 = access to AV)
A_0^τ	Set of links a not belonging to the AV-ready subnetwork in stage τ
A_1^τ	Set of links a belonging to the AV-ready subnetwork in stage τ
A	Set of all links a in the network;$A = \{A_0^\tau \cup A_1^\tau\}, \forall \tau \in \mathrm{T}$
Parameters	
μ_k	Logit route choice parameter for class k
θ_k	Logit mode choice parameter for class k
h_m	A constant representing attractiveness of mode m in the logit choice model
γ_m	PCE (PCU) of mode m
η_m	Value of travel time (VoTT) of mode m
$\overline{c}_{m,a}^\tau$	Fixed cost of mode m on link a in stage τ (i.e., driving cost for CVs and AVs, and fare for PT)
t_a^0	Free flow travel time on link a
α_a, b_a	BPR function parameters of link a
Λ_a	Capacity of link a
$\delta_{m,a}^{w,r,\tau}$	Assignment map: 1 if route r between OD pair w for mode m includes link a, 0 otherwise
e_m^τ	Yearly expenses of using vehicles of mode$m \in \{1,2\}$
σ	Parameter converting the hourly travel cost to a yearly value
ρ_n	Diffusion function scale parameters
κ_a^τ	Adjustment cost of link a in stage τ
ψ	Discount rate
$V^{\tau,w}$	Total number of vehicles (potential demand) between OD pair w in stage τ
$\widetilde{p}$	Market saturation rate of AVs
Variables	
$F_{m,r}^{\tau,w,k}$	(Route-based) flow of route r between OD pair w for class k and mode m in stage τ
$F_m^{\tau,w,k}$	Flow of mode m between OD pair w for class k in stage τ
$f_{m,a}^\tau$	(Link-based) flow of mode m on link a in stage τ
q_a^τ	(Link-based) total flow (PCE-equivalent) on link a in stage τ
$D^{\tau,w,k}$	Demand for (i.e., number of vehicles of) class k between OD pair w in stage τ
t_a^τ	(Link-based) Travel time on link a in stage τ
$c_{m,a}^\tau$	(Link-based) Travel cost of mode m on link a in stage τ
$C_{m,r}^{\tau,w,k}$	(Route-based) travel cost of route r between OD pair w for class k and mode m in stage τ
$\omega_m^{\tau,w,k}$	Expected satisfaction (logsum) of all routes for OD pair w for class k and mode m in stage τ
TTC^τ	Total system travel cost in stage τ
TTT^τ	Total system travel time in stage τ
TTD^τ	Total system travel distance in stage τ
TAC^τ	Total adjustment cost in stage τ
x_a^τ	Binary variable taking the value 1 if link a is upgraded in stage τ and 0 otherwise
X_a^τ	Binary variable with value of 1 if link a has been upgraded in stage τ or before and 0 otherwise
ncc_1^τ	Number of connected components in graph G_1^τ for stage τ

The travelers' response to these network design decisions is considered via a user equilibrium traffic assignment model including combined (simultaneous) choices of modes and routes, which are assumed to follow a hierarchical logit model. An AV diffusion model is used to estimate the market penetration rate of AVs in each stage based on the level of service (expected satisfaction) of all available travel choices in the previous stage and the price difference between CVs and AVs.

The following definitions are necessary to describe AV-ready subnetworks in mathematical terms. These definitions are demonstrated in Fig. 1.

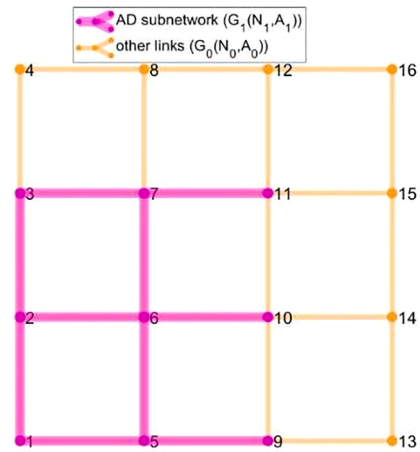


Fig. 1. An example AV-ready subnetwork; nodes 3–7-9–10-11 are boundary nodes, links (5,9)-(6,10)-(7,11) are inner boundary links, and links (9,13)-(10–14)-(11,15)-(11,12)-(7,8)-(3–4)-(9,10)-(10,11) are outer boundary links.

Definition 1. *Any node n_1 incident to at least one link included in an AV-ready subnetwork (i.e., $G_1(N_1, A_1)$), is included in that AV-ready subnetwork.*

Definition 2. *The degree of a node is the number of links incident to that node within the graph in which that node is included.*

Definition 3. *A boundary node n_1^* for an AV-ready subnetwork represented by a directed graph G_1 is a node included in $G_1$with a degree less than the degree of the corresponding node n in the underlying graph G (i.e., the graph representing the original road network).*

Definition 4. *An outer boundary link a_0^* for an AV-ready subnetwork is a link incident to a boundary node n_1^* of the AV-ready subnetwork but not included in the graph G_1 representing the AV-ready subnetwork.*

Definition 5. *An inner boundary link a_1^* for an AV-ready subnetwork is a link included in the graph G_1 representing the AV-ready subnetwork and incident only to boundary nodes with the degree of one.*

Note that boundary nodes of an AV-ready subnetwork and its inner boundary links are included in the subnetwork while the corresponding outer boundary links are not included in that subnetwork (hence the difference in the subscripts).

3.2. Lower level problem: combined multi-mode, multi-class stochastic user equilibrium

In the academic literature, the macroscopic static traffic assignment problem for CVs and AVs has been modeled using three general approaches. The first one is via the assumption of increased capacity for AVs (Chen et al., 2016; Ye and Wang, 2018). This approach is suitable when studying dedicated infrastructure for AVs. The next approach considers different routing principles for CVs and AVs (Bagloee et al., 2017; Wang et al., 2019). This is suitable when those principles (e.g., system optimal routing) are applicable for travel behavior of AVs. A combination of system optimal routing and increased capacity for AVs is used in Chen et al. (2017). Finally, the last approach is to consider AV-flow-specific (PCE-based) link travel times based on the ratio of AVs to CVs on each link along with assuming a different value of travel time (VoTT) for AD mode (Levin and Boyles, 2015; Liu and Song, 2019; Madadi et al., 2019). This approach is more common when considering mixed traffic conditions with CVs and AVs on links. It should be noted that system optimal routing is not realistic for mixed traffic conditions, particularly before reaching high market penetration rates of AVs. In this study, we follow the PCE-based approach and extend it for the multi-

mode traffic assignment problem under consideration.

Three travel modes, namely automated driving (AD) mode, conventional driving (CD) mode and public transport (PT) mode are considered here. Travelers are categorized within three separate classes based on their access to vehicles. Accordingly, each class has certain travel modes and their relevant routes available. Fig. 2 depicts the travelers' choice tree for each class. The summation of the demand of all modes for each class for each origin–destination (OD) pair is fixed for each stage, but it can change over time and the proportion of demand assigned to each mode can change based on the travelers' mode choice. The following equations represent the behavioral rules considered in the network equilibrium model (NEM) for each stage τ.

$$\sum_{w\in W}\sum_{k\in K}\sum_{r\in R_m^{w,\tau}} F_{m,r}^{\tau,w,k}\delta_{m,a}^{\tau,w,k} = f_{m,a}^{\tau}, \quad \forall a\in A,\ \forall\tau\in \mathrm{T},\ \forall m\in M, \tag{1}$$

$$\sum_{w\in W}\sum_{k\in K}\sum_{r\in R_m^{w,\tau}} C_{m,r}^{\tau,w,k}\delta_{m,r,a}^{\tau,w,k} = c_{m,a}^{\tau}, \quad \forall a\in A,\ \forall\tau\in \mathrm{T},\ \forall m\in M, \tag{2}$$

$$\sum_{r\in R_m^{w,\tau}} F_{m,r}^{\tau,w,k} = F_m^{\tau,w,k}, \quad \forall\tau\in \mathrm{T},\ \forall w\in W,\ \forall k\in K,\ \forall m\in M, \tag{3}$$

$$\sum_{m\in M} F_m^{\tau,w,k} = D^{\tau,w,k}, \quad \forall\tau\in \mathrm{T},\ \forall w\in W,\ \forall k\in K, \tag{4}$$

$$c_{m,a}^{\tau} = \overline{c}_{m,a}^{\tau} + \eta_m t_a^{\tau}, \quad \forall a\in A,\ \forall\tau\in \mathrm{T},\ \forall m\in M, \tag{5}$$

$$t_a^{\tau} = t_a^0\left[1+\alpha_a\left(\frac{q_a^{\tau}}{\Lambda_a^{\tau}}\right)^{b_a}\right], \quad \forall a\in A,\ \forall\tau\in \mathrm{T}, \tag{6}$$

$$q_a^{\tau} = \sum_{m\in M} f_{m,a}^{\tau}, \quad \forall a\in A_0^{\tau},\ \forall\tau\in \mathrm{T}, \tag{7}$$

$$q_a^{\tau} = \sum_{m\in M} \gamma_m f_{m,a}^{\tau}, \quad \forall a\in A_1^{\tau},\ \forall\tau\in \mathrm{T}, \tag{8}$$

$$F_{m,r}^{\tau,w,k}, F_m^{\tau,w,k} \geqslant 0, \quad \forall\tau\in \mathrm{T},\ \forall w\in W,\ \forall k\in K,\ \forall m\in M,\ \forall r\in R_m^{w,\tau}, \tag{9}$$

$$F_{1,r}^{\tau,w,0}, F_{2,r}^{\tau,w,0}, F_{2,r}^{\tau,w,1}, F_{1,r}^{\tau,w,2} = 0, \quad \forall\tau\in \mathrm{T},\ \forall w\in W,\ \forall r\in R_m^{w,\tau}. \tag{10}$$

Eq. (1) stablishes the correspondence between route and link flows, constraint (2) carries out the same for route costs and link costs. Eqs. (3) and (4) guarantee flow conservation for routes and modes, respectively. Constraint (5) represents link travel costs, which include link travel time multiplied by VoTT for each class, plus a fixed cost (e.g., per kilometer driving cost for cars and fare for PT).

Constraint (6) shows how link travel time is calculated based on a bureau of public roads (BPR) travel time function. This function calculates the link travel time using the total flow of vehicles on the link. On regular links (A_0^{τ}), the total flow is the simple summation of all flows on the link (Eq. (7)). However, on AV-ready subnetwork links (A_1^{τ}), the total flow is the weighted sum of the mode flows and their corresponding PCE value (Eq. (8)). The PCE value for the AD mode is lower than the PCE value for the CD mode to account for the shorted driving gaps between AVs and their leading vehicles in AD mode. As mentioned earlier, this approach is commonly used in the literature to calculate link travel times where AVs can use the AD mode in mixed traffic conditions. The exact PCE values used in this study are reported in Section 4.1.

Constraint (9) guarantees feasible flows, and (10) restricts flows of classes in modes unavailable to them (i.e., owners of CVs do not have access to AD mode, owners of AVs drive in AD mode whenever available, and the class with no vehicle available does not have access to any car mode).

In the following part, the equilibrium conditions of the NEM will be expressed in variational inequality (VI) formulation.

3.2.1. Equilibrium conditions

Let the set Π defined by (1)–(10) denotes the admissible set of flows for the NEM under consideration (i.e., Π defines the feasible region of the NEM). It should be noted that Π is a non-empty, convex and compact set since the demand is non-zero and finite. Then the equilibrium condition for the VI problem is to find $\pi^* = [F_{m,r}^{*\tau,w,k}, F_m^{*\tau,w,k}] \in \Pi$, such that $H(\pi^*)^T(\pi-\pi^*)\geqslant 0, \forall\pi\in\Pi$, where

$H(\pi) = [C_{m,r}^{\tau,w,k} + \frac{1}{\mu_m^k} lnF_{m,r}^{\tau,w,k}, \frac{1}{\theta_m^k} lnF_m^{\tau,w,k} - h_m^{\tau,w,k}]$, i.e., the VI problem is to solve

$$\begin{aligned}&\sum_{w\in W}\sum_{k\in K}\sum_{m\in M}\sum_{r\in R_m^{w,\tau}}[C_{m,r}^{*\tau,w,k}(F_{m,r}^{*\tau,w,k}) - \frac{1}{\mu_m^k} lnF_{m,r}^{*\tau,w,k}](F_{m,r}^{\tau,w,k} - F_{m,r}^{*\tau,w,k}) \\ &+\sum_{w\in W}\sum_{k\in K}\sum_{m\in M}[\frac{1}{\theta_m^k} lnF_m^{\tau,w,k} - h_m^{\tau,w,k}](F_m^{\tau,w,k} - F_m^{*\tau,w,k})\geqslant 0\end{aligned}, \tag{11}$$

subject to (1)–(10).

3.3. AV diffusion model

Since a long planning horizon including multiple time stages is considered in this study, the demand for AVs is likely to change over time. Therefore, this change needs to be estimated for each period based on the price of AVs and the benefits they provide. We use a diffusion model that estimates the AV demand in each period based on the mentioned factors. After each stage, the AV diffusion model is used to estimate the market penetration rate of AVs in the next stage endogenously based on the level of service (i.e., expected satisfaction) of all available choices to AVs in the previous stage and the price difference between CVs and AVs. The diffusion model takes the values of equilibrium travel cost ($C_{m,r}^{*\tau,w,k}$) obtained from solving (11) for each period τ as input and returns the demand for AVs for each OD pair w for the next period ($D^{\tau+1,w,2}$) as output to be used in the next stage.

Diffusion models have been widely utilized in various fields as

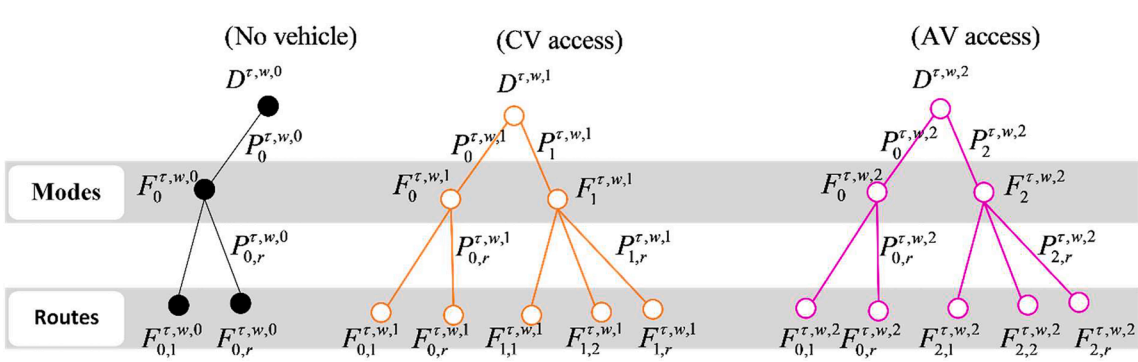


Fig. 2. Travel decision choice tree for three classes of travelers: from left, respectively, PT users (i.e., travelers without access to a vehicle ($k = 0$)), CV users ($k = 1$), AV users ($k = 2$).

models to predict market penetration rates of new products and technologies. Recently, they have been used to predict the adoption rate of AVs (Chen et al., 2016; Lavasani et al., 2016; Nieuwenhuijsen et al., 2018). In this study, we use an adaptation of the model proposed by Yang and Meng (2001) and used by Chen et al. (2016). This model is suitable for our study since it relates the adoption rate of AVs in each stage to its adoption rate in the previous stage, net benefits provided by them during the previous stage, and their cost. This allows time-varying estimation of the number of AVs in each stage endogenously using the utility values obtained by the discrete choice model used for the lower level problem. Therefore, the number of AVs adopted by each OD pair in each stage is calculated as follows.

$$D^{\tau+1,w,2} = D^{\tau,w,2} + g(\phi^{w,\tau})D^{\tau,w,2}\left(1 - \frac{D^{\tau,w,2}}{\tilde{p}V^{\tau+1,w}}\right), \quad \forall\tau \in \mathrm{T}, \quad \forall w \in W, \tag{12}$$

$$g(\phi^{w,\tau}) = \rho_1 exp(\rho_2[\phi^{w,\tau} - \overline{\phi}^{w,\tau}]), \quad \forall\tau \in \mathrm{T}, \quad \forall w \in W, \tag{13}$$

$$\phi^{w,\tau} = \sigma(\omega_1^{\tau,w,1} - \omega_2^{\tau,w,2}) - (e_1^{\tau} - e_2^{\tau}), \quad \forall\tau \in \mathrm{T}, \quad \forall w \in W, \tag{14}$$

$$\omega_m^{\tau,w,k} = \frac{1}{\mu_m^k} ln \sum_{r \in R_m^{\tau,w}} exp(\mu_m^k C_{m,r}^{*\tau,w,k}), \quad \forall\tau \in \mathrm{T}, \quad \forall w \in W, \quad \forall k \in K, \ \forall m \in M, \tag{15}$$

$$D^{\tau+1,w,1} + D^{\tau+1,w,2} = V^{\tau+1,w}, \quad \forall\tau \in \mathrm{T}, \quad \forall w \in W. \tag{16}$$

Eq. (12) shows how AV demand in each stage is dependent on the AV demand in the previous stage, total number of vehicles, market saturation rate of AVs, and the function $g(\phi^{w,\tau})$. This function represents the intrinsic growth coefficient for the OD pair w. Eqs. (13) and(14) show how this coefficient is calculated based on the price difference between CVs and AVs and the difference in benefits (i.e., expected satisfactions) provided by each vehicle type based on the utilities of all routes available for that vehicle type during the previous stage. Eq. (15) indicates how the expected satisfactions are calculated according to the utility values obtained from the route choice model ($C_{m,r}^{*\tau,w,k}$ is obtained from solving (11)). Eq. (16) guarantees conservation of the total number of vehicles in each stage. This number is a model input. Note that the total number of travelers with and without access to cars for each OD pair throughout the planning horizon ($V^{\tau,w}$ and $D^{\tau,w,0}$) are assumed to be fixed here. Nevertheless, a general or an OD-based growth rate (in case that information becomes available) can easily be applied using a multiplier in (16).

3.4. Upper level problem: multi-stage design of automated-vehicle-ready subnetworks

The upper level problem involves deciding which links to upgrade and include in the AV-ready subnetwork in each decision stage. The objective is to minimize the sum of total discounted adjustment cost and total discounted travel cost over the whole planning horizon given the travelers' response to each network configuration captured by the NEM (i.e., the flow patterns used in the upper level problem to calculate the objective function value are obtained from solving the NEM). The following mathematical program represents the upper level problem.

Min

$$Z_U = \sum_{\tau \in [0,\mathrm{T}]} \left\{ \frac{\sigma TTC^{\tau} + TAC^{\tau}}{(1+\psi)^{l\tau}} + \sum_{j \in [1,l-1]} \left\{ \frac{\sigma TTC^{\tau}}{(1+\psi)^{l\tau+j}} \right\} \right\}, \quad l \geqslant 2, \tag{17}$$

s.t.

(1)-(16),

$$TTC^{\tau} = \sum_{a \in A} \left\{ (1 - X_a^{\tau})[(\overline{c}_a^{\tau,1} + \eta^1 t_a^{\tau})(f_a^{*\tau,1} + f_a^{*\tau,2})] + X_a^{\tau}[(\overline{c}_a^{2,k} + \eta^2 t_a^{\tau}) f_a^{*\tau,2}] \right\}, \forall\tau \in \mathrm{T}, \tag{18}$$

$$TAC^{\tau} = \sum_{a \in A} x_a^{\tau} \kappa_a^{\tau}, \quad \forall\tau \in \mathrm{T}, \tag{19}$$

$$X_a^{\tau} = \sum_{\tau \in \mathrm{T}} x_a^{\tau}, \forall\tau \in \mathrm{T}, \quad \forall a \in A, \tag{20}$$

$$X_a^{\tau} \leqslant 1, \forall\tau \in \mathrm{T}, \quad \forall a \in A, \tag{21}$$

$$ncc_1^{\tau} = 1, \quad \forall\tau \in \mathrm{T}, \tag{22}$$

$$x_a^{\tau} \in \{0,1\}, \forall\tau \in \mathrm{T}, \quad \forall a \in A. \tag{23}$$

The objective function in (17) includes total discounted travel cost and total discounted adjustment cost. Since the length of each stage can be more than one year, a second term is added to the objective function to represent the net present value of the travel costs for years without any investment. Flows used in (18) are obtained by solving the NEM. Eq. (19) represents total adjustment cost in each stage, (20) and (21) ensure that each link can be upgraded only once, and after that point, it remains part of the subnetwork, and (23) shows the decision variables of the upper level problem are binary. They assume the value of one for links that are included in the subnetwork and zero otherwise (i.e.,$X_a^{\tau} = 1$, $\forall a \in A_1^{\tau}$). Constraint (22) denotes the connectivity requirement of the AV-ready subnetwork. It entails that for any two nodes within a subnetwork, there should be at least one path within the (undirected) graph representing that subnetwork that connects those two nodes. This means that the number of connected components in each subnetwork graph in each stage (ncc_1^{τ}) should be equal to one. The reason for inclusion of this constraint is to avoid subnetworks with separated parts. Since activating automated driving mode is allowed only on AV-ready subnetwork links, having subnetworks with separate components (i.e., disconnected subnetworks) leads to switching frequently between manual and automated mode, which should be avoided. This constraint imposes extra requirements on the solution methods, which are discussed in the following section.

3.5. Solution methods

We solve the problem introduced in this study using three evolutionary algorithms that are designed to deal with the complexity of the problem and the requirements mentioned earlier. We present two new algorithms, namely an evolutionary greedy search (EGS) and an evolutionary policy search (EPS), which were specifically developed for this problem. They deal with the connectivity constraint via tailored operations that are inspired by evolutionary processes, yet adjusted to preserve connectivity of subnetworks. Since GAs are one of the most common algorithms used in the literature to solve NDPs, we use a modified GA as a benchmark to compare the performance of our proposed algorithms. The GA used in this study copes with the connectivity constraint via a penalty function. All three algorithms use the so-called iterative-optimization-assignment approach where the upper level optimization problem and the lower level SUE assignment problem (NEM) are solved iteratively. After each stage, the diffusion model is executed once to estimate the AV demand for the next stage based on the equilibrium values of the previous stage. These three algorithms are described in detail in the following subsections.

3.5.1. Genetic algorithm (GA)

As mentioned earlier, the GA introduced in Holland (1975) and elaborately discussed in Golberg (1989), is one of the most successful heuristic algorithms used for solving DNDPs. Therefore, we have used it here as a benchmark for the performance of existing solution methods. However, some modifications were necessary to guarantee the connectivity requirement of AV-ready subnetworks (constraint (22)) and to accommodate the extra dimension (time) in decision variables. Instead of T sets of binary decision variables (one set for each stage), one set of

integer decision variables in range of $[1, T+1]$ is used where the value specifies in which stage the link will be upgraded ($T+1$ represents never). The connectivity constraint is dealt with via a penalty function in fitness evaluation. The GA operations are explained below and the GA procedure is shown in Table 2.

3.5.1.1. GA initialization. The GA introduced here is initialized with a random configuration (i.e., a random selection of values within the feasible range for each upper level decision variable).

3.5.1.2. GA mutation operation. A uniform mutation function is applied where 1% of genes (i.e., upper level decision variables) in each chromosome (i.e., solution vector) selected based on a uniform probability distribution are perturbed to generate mutated offspring.

3.5.1.3. GA crossover operation. Crossover operation here is based on a multiple-point crossover function where multiple (uniformly selected) swapping points for chromosomes are used to exchange genes between crossover parents. This means each gene in each crossover offspring has an equal chance of being inherited from either crossover parent.

3.5.1.4. GA fitness evaluation. Fitness evaluation in GA introduced here is based on the value of Z_U^{GA} in Eq. (24) where the value of Z_U is according to the Eq. (17), and the second term is a penalty function described in this section.

$$Z_U^{GA} = Z_U + \sum_{\tau \in [0,T]} 10^6 (ncc_1^\tau - 1) \tag{24}$$

The value of Z_U is based on the equilibrium link flows obtained from solving the NEM (lower level problem). Zhou et al., (2009) have established the existence and uniqueness of the solution for the VI formulation of the combined SUE problem (origin, destination, mode and route choice) with asymmetric link costs, which is a more general form of the NEM considered here. A comprehensive discussion on different formulations of the NEM and existing solution methods is provided in Florian and Hearn (1995). Common solution methods for the VI formulation of the NEM can be used to solve the lower level problem of the AD-NDP-T introduced here. Regarding algorithms to solve VIs, the literature is vast. For review studies, see Florian and Hearn (1995) and Harker and Pang (1990). In this study, we use a linear approximation type algorithm introduced in Wu et al. (2006) with step sizes according to the method of successive averages (MSA) and a parallel approach for updating class flows in each iteration to solve the NEM. The algorithm was used by Wu et al. (2006) to solve a multi-class VI problem with asymmetric link costs where the efficiency and convergence of the algorithm were shown on a large-scale network with 99,867 links.

In order to guide GA to find connected designs (i.e., satisfy constraint (22)), a penalty term is included in (24) for each subnetwork in each stage with more than one connected component (i.e., designs that violate constraint (22) are penalized in their fitness value). Note that there is exactly one connected component in any connected graph. Therefore, a penalty proportional to the number of extra components is imposed on designs with more than one connected component. This procedure guides GA to find designs with exactly one connected component. The value of the penalty weight in Eq. (24) is determined empirically and discussed in Section 4.1.2.

Each fitness evaluation includes solving the NEM and checking for connectivity T times (once per each stage) as well as running the diffusion model once after each stage to specify the AV demand for the next stage based on the travel utilities of the previous stage.

3.5.2. Evolutionary greedy search (EGS)

The EGS algorithm introduced in this study operates only on boundary links to preserve connectivity of produced subnetworks. It starts simple (one gene only) and the evolution process gradually adds complexity to the designs until there is no more gain from increasing complexity. To follow the general dynamic programming terminology, here we refer to an action as a decision for a single stage and a policy as a series of decisions for the complete planning horizon. EGS operates on action space. It starts from the first stage and optimizes the design for that stage. Once no further improvement is possible for that stage, it moves to the next stage. EGS operations are explained below and EGS procedure is shown in Table 3.

3.5.2.1. EGS initialization. EGS starts with a population of single links each one selected based on a roulette wheel prioritizing link capacity. Since there is no link elimination operation in EGS (i.e., once a link is added to the design, it stays in the design), starting with a larger number of links was found to be ineffective. Single links are sampled from the full set of feasible links to provide a sufficiently diverse (and connected) population pool to start the algorithm.

3.5.2.2. EGS extension operation. EGS operates on action space (i.e., optimizes one stage at a time). This means EGS extension operation is performed only for one stage at a time. Therefore, for each extension operation, first, the outer boundary links of the existing design for the active stage are found. Next, a sample of outer boundary links are selected based on a roulette wheel with odds proportional to link capacity, and a number of them are added to the existing design. Adding only outer boundary links guarantees that the resulting designs are connected (i.e., they meet constraint (22)). The number of candidate designs to evaluate (sample size) and the number of links to add to each

Table 2
GA procedure.

GA steps	
1	Initialize population (GA initialization) and measure fitness
2	**For** each generation **j**
3	**For** each individual **i**
4	Perform GA mutation operation
5	Measure fitness
6	**End**
7	Select parents using a binary tournament selection based on fitness
8	Perform GA crossover operation
9	Measure fitness
10	Select next generation from existing generation and offspring based on fitness
11	**If** stopping criteria met: **Terminate**
12	**End**

Table 3
EGS procedure.

EGS steps	
1	Initialize population (EGS initialization) and measure fitness
2	**For** each stage **τ**
3	**For** each generation **j**
4	**For** each individual **i**
5	Perform EGS extension operation (on outer boundary links)
6	Measure fitness
7	**End**
8	Select parents using a binary tournament selection based on fitness
9	Perform EGS merging operation (on parents that can produce connected offspring)
10	Measure fitness
11	Select next generation from existing generation and offspring based on fitness
12	**If** stopping criteria met: **Go To** the next stage
13	**End**
14	**End**

candidate (extension size) are algorithm parameters.

3.5.2.3. EGS merging operation. To preserve diversity among the population of designs, and to expedite the search process, a merging operation is applied in EGS. It requires two parents who are first checked for compatibility. That is, the parents are checked to determine whether they have a common node in their designs. Since both parents are connected designs, if they have a node in common, the union of their links will produce a connected graph. Note that checking for common nodes is computationally trivial compared to checking for connectivity after each merging. Then, if they pass the check, a merged design is generated from their union. The fraction of the population to consider for merging in each generation (merging fraction) is an EGS parameter.

3.5.2.4. EGS fitness evaluation. EGS fitness evaluation is based on the Z_U value in (17), which is obtained by the MSA-based linear approximation algorithm discussed earlier. However, to reduce computation times and avoid unnecessary fitness evaluations, this operation is sliced into several pieces. In each stage, EGS starts with the optimal design obtained at the end of the previous stage and makes adjustments only on the active stage's designs. Therefore, EGS fitness evaluation only includes adding the fitness value of each design for the active stage to the value of the optimal fitness at the end of previous stage. In this manner, at the end of the planning horizon, the fitness values correspond to the objective function value in (17) while avoiding a large number of unnecessary fitness evaluations for inactive stages. It will be shown in the next section that this approach is computationally efficient.

3.5.3. Evolutionary policy search (EPS)

EPS algorithm operates on policy space and uses operations inspired by evolutionary processes yet tailored to the problem to evolve to fitter designs while preserving connectivity. These operations include a context-aware merging and two types of mutations on boundary links. EPS operations are explained below and EPS procedure is shown in Table 4.

3.5.3.1. EPS initialization. EPS population is initialized with connected networks generated by the extension (similar to EGS extension) process without fitness evaluation. The number of starting links is a parameter. This means for each individual design, the algorithm starts with a random link, and adds links from the outer boundary links based on the roulette wheel explained earlier until a predefined number of links is added. The process provides the algorithm with a diverse set of connected designs to start the evolution process with minimal computational effort.

Table 4
EPS procedure.

EPS steps	
1	Initialize population (EPS initialization) and measure fitness
2	**For** each generation **j**
3	**For** each individual **i**
4	Randomly select the active time stage τ^*
5	Perform EPS extension operation (on outer boundary links)
6	Measure fitness
7	Randomly select the active time stage τ^*
8	Perform EPS reduction operation (on inner boundary links)
9	Measure fitness
10	**End**
11	Select parents using a binary tournament selection based on fitness
12	Perform EPS merging operation (on parents that can produce connected offspring)
13	Measure fitness
14	Select next generation from existing generation and produced offspring based on fitness
15	**If** stopping criteria met: **Terminate**
16	**End**

3.5.3.2. EPS extension operation. Since EPS operates on policy space, each time the extension operation is performed, first a stage τ^* is randomly selected as the active stage for the operation. Then, an extension operation similar to EGS extension is performed on the active stage's design. To guarantee constraint (20), once a set of links is selected to be added to a design on stage τ^*, the same set is added to all following stages (i.e., $[\tau^* + 1, T]$) of that design. The number of designs to consider for extension (extension sample) and the number of links to add to each design (extension size) are algorithm parameters.

3.5.3.3. EPS reduction operation. This is the process of eliminating unwanted links from designs to obtain better designs. However, to preserve connectivity of designs, candidate links for elimination are selected only from among inner boundary links. The number of links to be eliminated from each design (reduction size) and the number of candidate designs to consider for elimination (reduction sample) are algorithm parameters. As in the extension operation, here the active stage is randomly selected. However, the difference is that for the reduction operation, after the selection of candidates, all preceding stages of candidate designs are modified accordingly. This guarantees satisfaction of constraint (20).

3.5.3.4. EPS merging operation. EPS merging is also similar to the EGS merging process including the check for nodes in common to ensure connected offspring. The difference is that first, an active stage is randomly selected for EPS merging operation. If a common node is found for candidate parents in their active stage's designs, then a merged design is generated from their union in the active stage. The same design is used for the following stages, and the fitter parent contributes to designs of previous stages for the merged offspring. The fraction of the population to consider for merging in each generation (merging fraction) is an EPS parameter.

3.5.3.5. EPS fitness evaluation. EPS fitness evaluation is based on the Z_U value in (17), which is obtained by the MSA-based linear approximation algorithm discussed earlier. Similar to GA, EPS measures the fitness once per stage per design, with a diffusion function run after each stage to define demand for the next stage. However, no connectivity check or penalty is used for EPS since its operations guarantee connectivity. The optimal designs generated by EPS are checked for connectivity once after the optimization and as it will be shown later, they always meet the connectivity requirements of this problem.

4. Case study and numerical results

4.1. Description: Amsterdam metropolitan region

We demonstrate the multi-stage AV-ready subnetwork optimization concept on a case study of the Amsterdam metropolitan region. The network and demand data are obtained from the VENOM model (Kieft, 2013), which is based on the real network and demand patterns of Amsterdam, and is commissioned by the Amsterdam metropolitan region (Metropoolregio Amsterdam). It includes 52,812 links, 19,734 nodes, and 10,124 OD pairs (aggregated from 3722 original transportation zones). The study area (shown in Fig. 3) includes 24,250 links and 6642 OD pairs.

Demand data and availability of cars (thereby total number of users in each class) are extracted from the calibrated demand matrices per mode in VENOM model for the year 2004. All transportation demand from, to, within and through the study area is included in the OD matrix and accounted for in the assignment, but network performance indicators are reported for the OD pairs within the study area. One morning peak hour on an average workday is modeled and a conversion rate of $\sigma = 10*12*30$ (30 days a month, 12 month a year, daily to peak hour travel time ratio of 10) is used to obtain yearly values for

Fig. 3. Amsterdam case study: study area and feasible links for AV-ready subnetwork.

optimization. In the numerical results reported, all terms in Eq.(17) are divided by σ to avoid working with very large numbers. The traffic pattern of the base case (as is) is depicted in Fig. 4.

Since the focus of this study is automated vehicles, and in order to reduce computation times, a simplified assignment for public transport is used where for each OD pair, one (artificial) link represents expected satisfaction of all available routes available with public transport. The data for calculating mentioned values is derived from Brands (2015), where the author has used the same network and demand data (VENOM model) for an NDP study with a focus on public transport.

Regarding the diffusion model, the starting AV penetration rate used is 5%, the potential market size for AVs (saturation rate) is 90%, and the annual cost difference between CVs and AVs is 2000 €.

Motorways, regional roads and main urban roads are considered as feasible links for the subnetwork, and (per kilometer) link adjustment costs used in the case study are 50,000 €/km for motorways, 75,000 €/km for regional roads and 100,000 €/km for main urban roads. The maintenance cost per stage is 5% of the adjustment cost, which is added to the link adjustment costs. Net present values of economic benefits (i.e., total travel cost savings) and costs (i.e., total adjustment cost) up to one stage after the planning horizon are calculated and added to the reference point. The effect of variations of adjustment costs from mentioned values is considered via the sensitivity analysis reported in the next section. Total number of links selected as feasible links for the AV-ready subnetwork is 5804 out of 52,812 links (shown in Fig. 3). This leads to a lower level problem (i.e., NEM) with 52,812 continuous decision variables and an upper level problem with 5804 binary decision variables (2^{5804} possible solutions) for each stage. Three different scenarios, described below, are considered to account for the effects of planning horizon and decision stage length.

Scenario 1: six stages with preselection: A 30-year planning horizon is considered for all scenarios. In scenario 1, the planning horizon is divided into six stages (i.e., six viewing or decision points) with the length of five years for each stage. It is assumed in this scenario that for stage zero (corresponding to the beginning of the planning horizon), all motorways are (pre)selected as part of the AV-ready subnetwork (i.e.,

Fig. 4. Traffic patterns in Amsterdam case study: base case (as is) (bandwidth represents relative flow).

the value of decision variables for stage zero are preselected and fixed). The main reason for this choice is that many experts believe motorways are the first places to facilitate automated driving; therefore, they should be included in any network configuration for AVs. Moreover, reducing the number of decision variables (of the upper level problem) can decrease computation times as well.

Scenario 2: two stages with preselection: In this scenario, a 30-year planning horizon with two stages (each 15 years long) and the same link preselection for stage zero as scenario 1 is considered. Since in this scenario the length of the planning horizon is the same as scenario 1 yet investment decisions are made less frequently, the comparison aids in demonstrating the effects of investment frequencies.

Scenario 3: two stages without preselection: A 30-year planning horizon with two time stages (each 15 years long) and no link preselection for stage zero is considered in this scenario. Since the value of the objective function for stage zero is also included in all calculations for all scenarios, comparing scenario 3 with scenario 2 provides some insight into the impacts of preselecting motorways.

4.1.1. Hardware and software

The mathematical model and the solution algorithms were coded in MATLAB and ran on a Windows PC with a Core i5-8600 CPU @ 3.10 GHz and 32 GB RAM. MATLAB parallel computation toolbox was utilized with six parallel computing units for efficient computations and dealing with the computational complexity arising from the problem size. It should be noted that population-based algorithms (such as the evolutionary algorithms used in this study) can fully utilize the potential of parallel computation for efficiency. Moreover, sparse matrices in MATLAB were used for all algebraic operations on assignment maps to minimize the computation times of the MSA-based algorithm used to solve the lower level problem (NEM). Overall, 3 scenarios, 3 algorithms and 5 runs for each algorithm in each scenario led to a total of 45 optimization runs for the Amsterdam case study (excluding parameter tuning and sensitivity analysis), which culminated in approximately 72 days of computations.

4.1.2. Parameter tuning

For computational experiments of the Amsterdam case study, run time limits (i.e., maximum computation times) of 10 h for scenario 2 and scenario 3, and 160 h for scenario 1 were considered. We experimented with a grid of different parameter values for each algorithm and conducted the final computational experiments using the parameters that led to the best results (lowest objective function values) for each algorithm within the mentioned time limits. The best performing parameters that were used for the reported experiments for all three algorithms are shown in Table 5. Regarding the penalty weight used for GA in Eq. (24), the value of 10^6 is selected optimally after experimenting with different values in range of $[10, 10^7]$. A summary of these experiments for the three scenarios in this study is illustrated in Fig. 5.

Table 5
Best performing parameter values for EGS, EPS and GA in Amsterdam case study.

EGS		EPS		GA	
Parameter	**Value**	**Parameter**	**Value**	**Parameter**	**Value**
Population size	30	Population size	300	Population size	180
Extension size	5	Extension size	5	Max generations	2000
Sample size	8	Reduction size	2	Elite size	12
Merging fraction	0.8	Extension sample	2	Crossover fraction	0.9
		Reduction sample	2	Crossover type	uniform
		Merging fraction	0.9	Mutation fraction	0.01

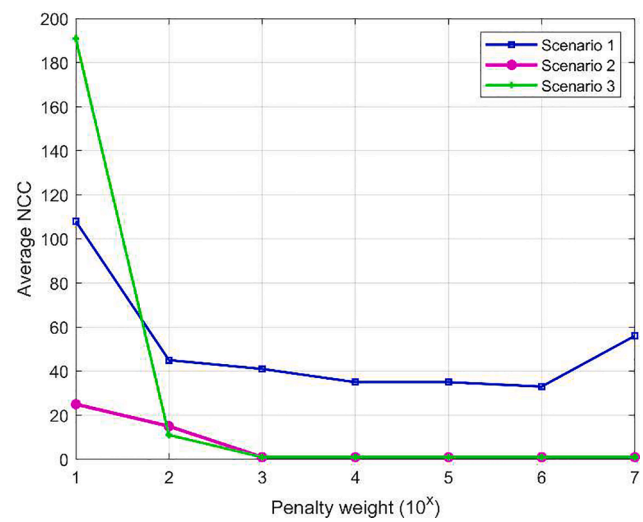


Fig. 5. Number of connected components in GA results for different penalty weight values in all scenarios.

Table 6
Lower bounds and upper bounds for OF, TTC and TAC values.

	OF (€)	TTC (€)	TAC (€)
Scenario 1			
Upper bound	23,014,977	1,185,655	256,024,217
Lower bound	22,457,536	1,145,451	0
Scenario 2			
Upper bound	25,549,484	1,185,655	216,115,436
Lower bound	24,972,031	1,143,857	0
Scenario 3			
Upper bound	25,549,484	1,185,655	209,521,104
Lower bound	24,989,569	1,143,907	0

4.2. Numerical results and analysis

In this section, we assess the effects of deploying AV-ready subnetworks on network performance using three main network performance criteria, namely, total travel cost (TTC), total travel time (TTT) and total travel distance (TTD). Reported values for objective function (OF) in Tables 6–8 are averages of five independent runs (replications). The bandwidths for 95% confidence intervals are reported in Table 7. Computation times (CTs) are reported in the same manner. TTC, TTT and TTD values are averages per stage per run. This allows comparisons with the base case. Note that due to the scaling of Eq. (17) explained in case study description, the reported values of TTC, TTT and TTD are hourly values rather than yearly values. Total adjustment cost (TAC) values are summed over all stages and averaged over all runs to show total investment values per scenario.

Moreover, we discuss the performance of the solution algorithms and the impacts of scenario settings with respect to several performance criteria. The solution algorithms are compared in terms of effectiveness (i.e., average OF value obtained), efficiency (i.e., average computation times), stability (i.e., within run variations), scalability (i.e., computation time to problem size ratio), and constraint satisfaction (i.e., producing connected graphs).

4.2.1. Lower bounds

Since the global optimum point of the problem is unknown, and with heuristics, there is no guarantee to find this point, we have provided additional numbers as reference points for comparisons. First, the values

Table 7
Summary of case study results (±signs denote standard errors for 95% confidence intervals).

Algorithm	OF (€) (average)	TTC (€) (average)	TTT (h) (average)	TTD (km) (average)	TAC (€) (sum)	Connected	CT(h) (average)
***Scenario 1:** six stages with preselection*							
GA	22,597,906 ± 3,635	1,150,222	97,140	1,373,965	245,869,501	No	154.92 ± 43.10
EGS	22,534,825 ± 33,797	1,146,307	97,051	1,378,132	250,380,768	Yes	5.53 ± 0.14
EPS	22,685,967 ± 1,626	1,156,838	97,289	1,366,779	240,203,049	Yes	144.06 ± 0.97
As is	23,014,977	1,185,655	97,208	1,314,145	0	–	0
Scenario 2: two stages with preselection							
GA	25,143,063 ± 321	1,150,777	97,181	1,373,674	212,675,527	Yes	7.95 ± 0.08
EGS	25,023,282 ± 3	1,144,292	97,026	1,380,753	216,024,677	Yes	4.03 ± 0.03
EPS	25,139,950 ± 333	1,150,612	97,175	1,373,890	212,680,670	Yes	9.15 ± 0.09
As is	25,549,484	1,185,655	97,208	1,314,145	0	–	0
Scenario 3: two stages without preselection							
GA	25,282,900 ± 243	1,159,160	97,106	1,357,159	198,926,645	Yes	8.78 ± 0.15
EGS	25,002,051 ± 1,138	1,144,221	96,920	1,380,716	131,099,802	Yes	6.72 ± 0.18
EPS	25,029,462 ± 1,400	1,145,238	97,026	1,379,269	209,199,259	Yes	3.78 ± 0.17
As is	25,549,484	1,185,655	97,208	1,314,145	0	–	0

Table 8
Percentage of optimality gaps (between upper bounds and lower bounds) achieved by algorithms.

	OF (€)	TTC (€)	TTC saving (€) *	TAC saving (€)**	TAC spending (€)***
Scenario 1					
GA	75%	88%	127,558,800	10,154,716	8,195,650
EGS	86%	98%	141,652,800	5,643,449	8,346,026
EPS	59%	72%	103,741,200	15,821,168	8,006,768
Scenario 2					
GA	70%	83%	125,560,800	3,439,909	7,089,184
EGS	91%	99%	148,906,800	90,759	7,200,823
EPS	71%	84%	126,154,800	3,434,766	7,089,356
Scenario 3					
GA	48%	63%	95,382,000	10,594,459	6,630,888
EGS	98%	99%	149,162,400	78,421,302	4,369,993
EPS	93%	97%	145,501,200	321,845	6,973,309

* TTC saving values reported are yearly and in comparison with the upper bounds reported in Table 6.
** TAC saving values reported are in comparison with the upper bounds reported in Table 6.
*** TAC spending values reported are yearly.

of the base case scenario (as is) are calculated to evaluate the network performance as it is before the changes. This provides the upper bounds for the values of OF and TTC. Second, the TTC values for a variation where all feasible links are included in the AV-ready subnetwork in all stages are calculated to provide a lower bound for TTC values obtained by deployment of AV-ready subnetworks. Since OF is the summation of TTC and TAC, using mentioned lower bounds for TTC values along with the value of zero for TAC (assuming no adjustment cost for reaching these TTC values) in each scenario provides (underestimated) lower bounds for the OF values. We use mentioned lower bounds as well as the values of the "as is" scenario and maximum possible investment costs to report the lower and the upper bounds for the values of OF, TTC and TAC in Table 6. The "as is" values of all performance indicators are reported in Table 7. These values aid in evaluating the case study results reported in Table 7 and calculating the optimality gaps reported in Table 8. It should be noted that although the lower bounds for OF might be unachievable in practice due to their underestimation, it is guaranteed that no algorithm can achieve lower numbers for OF values. This is a desirable property for algorithm performance comparison purposes, and it is shown in Table 8 that results very close to the lower bounds can be achieved.

4.2.2. Network performance

As evidenced by Table 7, there is a notable network-wide decrease in (per-stage) TTC values in all scenarios with all algorithms compared to the base case, which indicates that the AV-ready subnetwork concept has an overall positive impact on network performance in terms of TTC. In addition, TTC values (especially for EGS) are very close to the lower bounds reported in Table 8. The value of the TTC gap between the upper and the lower bound is above 98% for EGS in all scenarios. The OF gap values achieved by EGS are between 86% and 98%. Given that the lower bounds reported for OF values are underestimated, these results suggest that the results (at least for EGS algorithm) are near optimal.

Regarding TTT, mild improvements are observed in all cases (with the exception of EPS in scenario 1). However, these improvements are not as significant as the observed TTC improvements. The differences between TTC and TTT in the extent of improvements were expected, since the algorithms optimize for TTC and TAC but not for TTT. Moreover, the tendency to choose longer routes via the AV-ready subnetworks in automated mode can cause more travel time while leading to lower TTC due to the lower VoTT and higher fuel efficiency of automated driving. This change in travelers' route choice behavior caused by the deployment of AV-ready subnetworks has been observed previously (Madadi et al., 2020). It also explains considerably higher TTDs in all optimal cases compared to the base case.

The traffic patterns of optimal cases are very similar to the base case patterns (Fig. 4) with a slight increase in the volumes in the latter case. The explanation is that when AVs become more attractive in time with the existence of the AV-ready subnetworks, more travelers opt for cars. This leads to an increase in volumes on the network; nonetheless, the performance of the network is still favorable to the base case in terms of TTC and TTT due to the efficiency of automated driving.

4.2.3. Algorithm performance

In terms of effectiveness (i.e., average OF value obtained) EGS shows a better performance compared to the competing algorithms in all scenarios, particularly in scenario 1 when the number of possible solutions is considerably higher. This is evidenced by the numbers reported in Table 8 where EGS covers larger proportions of optimality gaps. In scenario 1 where the problem becomes very large, the greediness of EGS seems to serve well in terms of both efficiency and effectiveness; although, the optimality gap reduced by EGS is lower in this scenario compared to other scenarios. GA on the other hand, is outperformed in effectiveness in all scenarios by EGS; however, in scenario 1, it outperforms EPS, and in scenario 2, it has a very similar performance to EPS.

When it comes to efficiency (i.e., average computation times), EGS performs well on average yet not as well as its effectiveness performance. For instance, in scenario 3, EPS has a lower computation time compared to EGS. EGS starts with a single link, and adds only a small number of links at each generation. Therefore, in scenario 3 where there

is no link preselection, which leads to a higher number of decision variables, it takes longer than other scenarios, even longer than scenario 1.

As for stability, we reflect on within run variations captured by standard errors of both OF values and computation times (denoted by ±signs in Table 7). GA and EPS demonstrate notable stability in both OF values and computation times in scenario 2 and scenario 3. However, in scenario 1, EPS is somewhat stable around an undesirable OF value, and GA is highly unstable in terms of computation times. Conversely, EGS shows rather persistent stability in computation times. Regarding OF values, EGS has the highest stability in scenario 2, followed by scenario 3 and scenario 1. It is worth noticing that although EGS shows a rather large within run variability for OF in scenario 1, its worse run is still superior to the other algorithms' best run (i.e., the upper bound of EGS confidence interval for OF is significantly lower that the lower bounds of GA and EPS confidence intervals for their OF value).

Regarding scalability (i.e., computation time to problem size ratio), computation times of EPS and GA grow exponentially and their performances deteriorate with the increase in the number of stages, while EGS computation times grow rather linear and remain effective with more stages. With the increase in the number of decision variables, EGS computation times grow rather linear while maintaining their level of effectiveness, whereas EPS and GA show less sensitivity to (a limited) increase in the number of decision variables. This relates to their structure and operation space; EPS and GA operate on policy space (i.e., one set of decision variables for the entire planning horizon), whereas EGS operates on action space (i.e., dealing with the decision variables of each time stage separately). In policy space, the number of possible solutions grows exponentially with the number of stages, while in action space, when the values of decision variables of previous stages are fixed in each stage, the number of possible solutions increases linearly with the increase in the number of stages. On the other hand, operating in such a manner on action space in a multi-stage setting makes EGS a greedy algorithm, since the decisions on each stage are taken without considerations for the later stages. Yet that does not seem to deter its performance in this problem. This could relate to the nature of the problem, since another greedy algorithm has been shown to perform well for the AV-ready subnetwork optimization problem without the time dimension (Madadi et al., 2020). Future studies should evaluate the performance of EGS on other problem instances and settings to investigate the effects of its greediness.

Constraint satisfaction is defined in this study as an algorithm's ability to satisfy constraint (22), i.e., produce connected AV-ready subnetworks as solutions. For a comprehensive discussion on the necessity of this constraint, the reader is referred to Madadi et al. (2020). As is shown in Table 7 and depicted in Figs. 6–8, EGS and EPS generate connected subnetworks in all stages and all scenarios. This was to be expected due to their operations, which are tailored to this purpose. On the contrary, GA does not always meet this criterion, especially in scenario 1. Even though a penalty is applied for each disconnected component in each stage, and various values for this penalty were explored. It is shown in Madadi et al. (2020) that GA with a similar penalty function successfully finds connected designs for the single stage, unimodal problem. Likewise, GA with penalty satisfies the connectivity constraint in scenario 2 and scenario 3 of this study. However, as the number of stages increase, the penalty function appears to be less effective for satisfying the connectivity constraint. This goes to show that the problem size is a defining factor for performance of the algorithms. Therefore, it should be taken into consideration while selecting an appropriate solution method for such problems.

4.2.4. Sensitivity analysis

In this section, we discuss possible variations in model input parameters and their impacts on the output. The values reported in Table 9 are averages of three independent EGS runs for scenario 3. We analyzed three general categories of parameters, namely, adjustment cost parameters, diffusion model parameters and mode choice parameters.

Based on Table 9, the most sensitive model parameter is ρ_1, which represents the sensitivity of AV demand to its overall cost (i.e., generalized travel cost and ownership cost) in the diffusion model. This is also evident from Fig. 9 where it has been demonstrated that the evolution of market penetration rate of AVs can take considerably different paths with different values of this parameter. This signifies the importance of accurate AV demand prediction for infrastructure planning decisions.

On the contrary, adjustment cost parameters are the least sensitive parameters. Although severe and rather proportional changes in TAC values are observed with variations in adjustment costs, which is natural, the changes in objective function values and network performance indicators are trivial in all cases. This indicates that adjustment cost parameters can have a significant impact on the project cost but not on network performance.

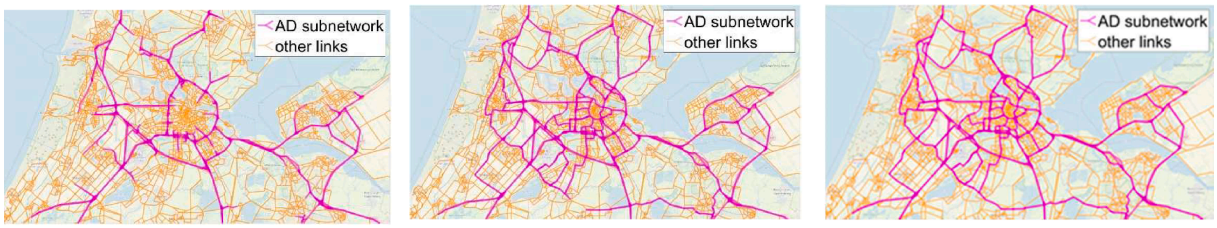
Regarding mode choice parameters, an increase in the (absolute) value of the sensitivity of AVs to the level of service (expected satisfaction) has the least significant impact on the results. On the other hand, a decrease in the sensitivity of AVs to the level of service demands greater investment with less positive impact on network performance. With CVs, the direction of changes is the opposite; less sensitivity to the level of service leads to better network performance and more sensitivity diminishes network performance.

5. Summary, conclusions and future research directions

In this study, we considered the problem of multi-stage optimization of AV-ready subnetworks within road networks with time-varying demand. We modeled the problem as a time-dependent bi-level NDP where the upper level denoted infrastructure decisions made by authorities in several stages over a planning horizon, and the lower level represented mode and route choices of different classes of travelers (with access to AVs, CVs or no vehicle) in each stage in response to the infrastructure supply. We presented the VI formulation of the lower level as a multi-class simultaneous mode and route choice UE using a hierarchical logit model, and solved it using a linear approximation type algorithm with step sizes based on MSA. The upper level problem was modeled as a mathematical program with binary decision variables and solved for near-optimal solutions using three evolutionary heuristics. It is crucial to notice that network configurations for AVs are sensitive to the level of

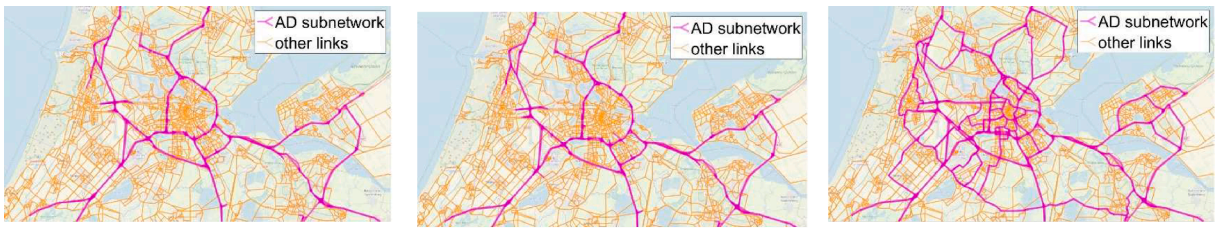
(a) Stage 2 (b) Stage 4 (c) Stage 6

Fig. 6. Optimal evolution of AV-ready subnetworks obtained by GA for Scenario 1.



(a) Stage 2 (b) Stage 4 (c) Stage 6

Fig. 7. Optimal evolution of AV-ready subnetworks obtained by EGS for Scenario 1.



(a) Stage 2 (b) Stage 4 (c) Stage 6

Fig. 8. Optimal evolution of AV-ready subnetworks obtained by EPS for Scenario 1.

Table 9
Sensitivity analysis (% changes in network performance criteria with model parameter variations).

Category	Parameter	Variation	OF	TTC	TTT	TTD	TAC
Adjustment cost		(−50%, −50%,-50%)	−0.04%	0.00%	+0.01%	−0.01%	−45.58%
	Cost ratios:	(+50%,+50%,+50%)	+0.04%	+0.01%	0.00%	−0.01%	+43.20%
	(Motorway, Regional, Urban)	(+50%, 0% , −50%)	+0.03%	+0.02%	0.00%	−0.01%	+17.51%
		(+100%, 0%,-100%)	+0.02%	−0.01%	−0.01%	+0.01%	+28.75%
Mode choice	θ_2	+50%	0.00%	+0.01%	−0.01%	+0.00%	−2.58%
	θ_2	−50%	+0.02%	+0.02%	+0.04%	−0.03%	+8.86%
	θ_1	+50%	+0.01%	+0.01%	+0.02%	−0.02%	+4.44%
	θ_1	−50%	−0.02%	−0.02%	−0.06%	+0.03%	−10.99%
Diffusion model	ρ_1	+50%	−0.69%	−0.83%	−0.03%	+1.15%	+4.59%
	ρ_1	−50%	+1.36%	+2.02%	+0.14%	−2.68%	−11.64%
	ρ_2	+50%	0.00%	0.00%	0.00%	0.00%	−2.36%
	ρ_2	−50%	+0.01%	+0.00%	+0.01%	−0.01%	+8.02%

AV demand, which is expected to evolve over time. Therefore, these configurations should evolve over time as well. This makes the multi-stage planning approach necessary, which adds tremendous complexity to the problem and calls for efficient solution methods. Furthermore, we used the real road network of the Amsterdam metropolitan region to demonstrate the concept and compare the performance of the solutions.

Two tailored evolutionary algorithms, namely EGS and EPS, were developed in this study and their performance was compared to a GA with a penalty function (to satisfy constraints) using the case study. Both EGS and EPS successfully satisfied the constraints in all scenarios while GA became less effective in meeting this requirement with larger number of stages. Regarding effectiveness, EGS showed a satisfactory performance in all scenarios considered in this study. As for efficiency, EGS outperformed other algorithms in two of the scenarios considered in this study; however, it was outperformed by EPS in the third scenario. The advantage of GA over the proposed algorithms is that it is available in most optimization packages and can easily be applied to the problem with a rather standard penalty function to satisfy constraints. However, in this study, with the growth in problem size, a decrease in the effectiveness of the GA was observed.

It was shown that AV-ready subnetworks deliver significant benefits in terms of TTC. The extent of these benefits increased with higher AV penetration rates and more AV-ready roads. However, this was accompanied by higher travel distances for AVs, which might cause higher emissions.

On the other hand, AVs are expected to be more fuel-efficient compared to CVs. Moreover, many vehicle manufacturers have started manufacturing or have announced plans to produce electric-powered AVs. This could offset the negative impacts of longer travel distances with AVs in terms of emissions. However, the extent to which the fuel-efficiency of AVs and the coupling of automation and electrification trends in car industry can counterbalance the environmental impacts of longer AV travel distances is uncertain.

A possible approach to optimize this trade-off is via monetizing the emission impacts and including the costs in the objective function of the network design problem alongside travel cost and infrastructure investment cost. Another alternative would be considering emissions, or travel distance as a proxy for emissions, as another objective and formulating a multi-objective network design problem.

Performance of the algorithms considered in this study was compared using a case study of the Amsterdam metropolitan region. Future research can focus on the application of these algorithms on other problem instances and settings for further investigation of the

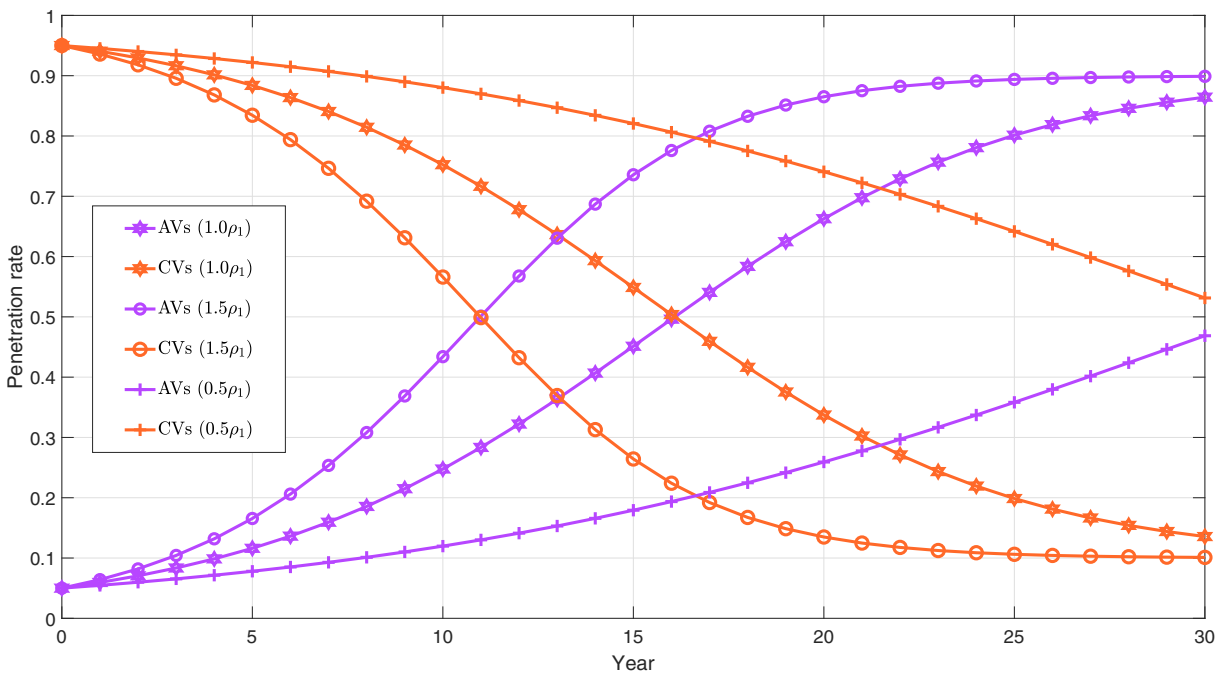


Fig. 9. Sensitivity analysis for diffusion model (scenario 3).

performance of these algorithms.

Macroscopic static traffic assignment models have been commonly used for the lower level of bi-level NDPs, even though dynamic traffic assignment models can capture the behavioral differences of CVs and AVs more accurately. This is due to the general complexity of NDPs and high computation times of dynamic traffic assignment models. With advances in technology, computers with higher computation power are becoming available. However, application of dynamic traffic assignments for AD-NDP-T studies, which are computationally much more demanding compared to standard NDP studies, remains a challenge, especially for case studies of large-scale networks.

An interesting extension of this study is combining other network configurations, such as dedicated lanes and roads for AVs with AV-ready subnetworks for mixed traffic and developing a unified modeling framework for combinations of network design concepts for AVs. So far, these concepts have been modeled separately with incompatible frameworks. This makes it difficult to study them simultaneously using one model. Nevertheless, their combination can be relevant, particularly for large regions with various road types and jurisdictions.

The sensitivity analysis performed in this study indicated that the AV diffusion model parameters are the most sensitive parameters of the study. Therefore, fine-tuning these parameters can aid in accurate estimation of AV market penetration rate over time. However, since highly automated vehicles are not available on the market yet, fine-tuning the diffusion model parameters or validating its output is not possible at the moment. Moreover, models that can estimate market penetration rate of AVs and are compatible with discrete choice models are still rare in the academic literature. Nonetheless, an ideal AV demand estimation model for NDPs with AVs should be less reliant on scale parameters.

When a long planning horizon is considered, origin and destination choices might become relevant as well. This can add yet another layer of complexity to an already complex model. Nonetheless, assuming fixed OD pairs is a non-trivial simplification present in all NDP studies with a few exceptions.

Finally, a more detailed multimodal traffic assignment model including active modes, a more detailed representation of public transport, and combined modes such as park and ride can capture all available choices to the travelers and improve the accuracy of the model results.

CRediT authorship contribution statement

Bahman Madadi: Conceptualization, Data curation, Formal analysis, Methodology, Software, Visualization, Writing - original draft. **Rob van Nes:** Conceptualization, Data curation, Formal analysis, Methodology, Supervision, Writing - review & editing. **Maaike Snelder:** Conceptualization, Data curation, Formal analysis, Methodology, Supervision, Writing - review & editing. **Bart van Arem:** Conceptualization, Data curation, Formal analysis, Methodology, Supervision, Writing - review & editing, Funding acquisition.

Acknowledgement

This study was funded as part of the project Spatial and Transport impacts of Automated Driving (STAD) by the Netherlands Organization for Scientific Research (NWO) under contract 438-15-16.

Data Availability

The MATLAB codes used for deployment of the mathematical model and the solution algorithms used in this study are available from the corresponding author upon request. The network and demand data of the case study of Amsterdam were extracted from the VENOM model (Kieft, 2013) ("Verkeerskundig Noordvleugel Model" in Dutch) under license and can be requested at Theo van der Linden; venom @ vervoerregio.nl.

References

Bagloee, S.A., Sarvi, M., Patriksson, M., Rajabifard, A., 2017. A Mixed user-equilibrium and system-optimal traffic flow for connected vehicles stated as a complementarity problem: mixed user-equilibrium and system-optimal traffic flow. Comput.-Aided Civ. Infrastruct. Eng. 32 (7), 562–580. https://doi.org/10.1111/mice.12261.

Brands, T., 2015. Multi-objective Optimisation of Multimodal Passenger Transportation Networks. University of Twente.

Cantarella, G.E., Pavone, G., Vitetta, A., 2006. Heuristics for urban road network design: lane layout and signal settings. Eur. J. Oper. Res. 175 (3), 1682–1695. https://doi.org/10.1016/j.ejor.2005.02.034.

Chen, M., Alfa, A.S., 1991. A network design algorithm using a stochastic incremental traffic assignment approach. Transp. Sci. 25, 215–224. https://doi.org/10.2307/25768493.

Chen, Z., He, F., Yin, Y., Du, Y., 2017. Optimal design of autonomous vehicle zones in transportation networks. Transp. Res. Part B: Methodol. 99, 44–61. https://doi.org/10.1016/j.trb.2016.12.021.

Chen, Z., He, F., Zhang, L., Yin, Y., 2016. Optimal deployment of autonomous vehicle lanes with endogenous market penetration. Transp. Res. Part C: Emerg. Technol. 72, 143–156. https://doi.org/10.1016/j.trc.2016.09.013.

Dantzig, G.B., Harvey, R.P., Lansdowne, Z.F., Robinson, D.W., Maier, S.F., 1979. Formulating and solving the network design problem by decomposition. Transp. Res. Part B: Methodol. 13 (1), 5–17. https://doi.org/10.1016/0191-2615(79)90003-1.

Davis, G.A., 1994. Exact local solution of the continuous network design problem via stochastic user equilibrium assignment. Transp. Res. Part B: Methodol. 28 (1), 61–75.

Dixit, V.V., Chand, S., Nair, D.J., 2016. Autonomous vehicles: Disengagements, accidents and reaction times. PLoS ONE 11, 1–14. https://doi.org/10.1371/journal.pone.0168054.

Farah, H., Erkens, S.M.J.G., Alkim, T., van Arem, B., 2018. Infrastructure for automated and connected driving: state of the art and future research directions. In: Beiker, G., Meyer, S. (Eds.), Road Vehicle Automation 4, Lecture Notes in Mobility. Springer International Publishing, pp. 187–197. https://doi.org/10.1007/978-3-319-60934-8.

Farahani, R.Z., Miandoabchi, E., Szeto, W.Y., Rashidi, H., 2013. A review of urban transportation network design problems. Eur. J. Oper. Res. 229 (2), 281–302. https://doi.org/10.1016/j.ejor.2013.01.001.

Favarò, F., Eurich, S., Nader, N., 2018. Autonomous vehicles' disengagements: trends, triggers, and regulatory limitations. Accid. Anal. Prev. 110, 136–148. https://doi.org/10.1016/j.aap.2017.11.001.

Fischetti, M., Ljubić, I., Monaci, M., Sinnl, M., 2018a. On the use of intersection cuts for bilevel optimization. Math. Program. 172 (1-2), 77–103. https://doi.org/10.1007/s10107-017-1189-5.

Fischetti, M., Ljubić, I., Monaci, M., Sinnl, M., 2017. A new general-purpose algorithm for mixed-integer Bilevel linear programs. Oper. Res. 65 (6), 1615–1637. https://doi.org/10.1287/opre.2017.1650.

Fischetti, M., Monaci, M., Salvagnin, D., 2018b. SelfSplit parallelization for mixed-integer linear programming. Comput. Oper. Res. 93, 101–112. https://doi.org/10.1016/j.cor.2018.01.011.

Florian, M., Hearn, D., 1995. Network equilibrium models and algorithms. Handb. Oper. Res. Manag. Sci. 8, 485–550. https://doi.org/10.1016/S0927-0507(05)80110-0.

Golberg, D.E., 1989. Genetic Algorithms in Search Optimization & Machine Learning. Addison-Wesley.

Hardman, S., Lee, J.H., Tal, G., 2019. How do drivers use automation? Insights from a survey of partially automated vehicle owners in the United States. Transp. Res. Part A: Policy Pract. 129, 246–256. https://doi.org/10.1016/j.tra.2019.08.008.

Harker, P.T., Pang, J.-S., 1990. Finite-dimensional variational inequality and nonlinear complementarity problems: a survey of theory, algorithms and applications. Math. Program. 48 (1-3), 161–220. https://doi.org/10.1007/BF01582255.

Holland, J.H., 1975. Adaptation in Natural and Artificial Systems. University of Michigan Press, Ann Arbor, MI.

Kieft, S., 2013. Handboek Venom (in Dutch). The Netherlands, Stadsregio Amsterdam.

Lachhwani, K., Dwivedi, A., 2018. Bi-level and multi-level programming problems: taxonomy of literature review and research issues. Arch. Computat. Methods Eng. 25 (4), 847–877. https://doi.org/10.1007/s11831-017-9216-5.

Lavasani, M., Jin, X., Du, Y., 2016. Market penetration model for autonomous vehicles on the basis of earlier technology adoption experience. Transp. Res. Rec. 2597 (1), 67–74. https://doi.org/10.3141/2597-09.

Leblanc, L.J., 1975. An algorithm for the discrete network design problem. Transp. Sci. 9 (3), 183–199. https://doi.org/10.1287/trsc.9.3.183.

Levin, M.W., Boyles, S.D., 2015. Effects of autonomous vehicle ownership on trip, mode, and route choice. Transp. Res. Rec. 2493 (1), 29–38. https://doi.org/10.3141/2493-04.

Li, Y., Ukkusuri, S.V., Fan, J., 2018. Managing congestion and emissions in transportation networks with dynamic carbon credit charge scheme. Comput. Oper. Res. 99, 90–108. https://doi.org/10.1016/j.cor.2018.06.014.

Liu, Z., Song, Z., 2019. Strategic planning of dedicated autonomous vehicle lanes and autonomous vehicle/toll lanes in transportation networks. Transp. Res. Part C: Emerg. Technol. 106, 381–403. https://doi.org/10.1016/j.trc.2019.07.022.

Lo, H.K., Szeto, W.Y., 2009. Time-dependent transport network design under cost-recovery. Transp. Res. Part B: Methodol. 43 (1), 142–158. https://doi.org/10.1016/j.trb.2008.06.005.

López-Ramos, F., Nasini, S., Guarnaschelli, A., 2019. Road network pricing and design for ordinary and hazmat vehicles: Integrated model and specialized local search. Comput. Oper. Res. 109, 170–187. https://doi.org/10.1016/j.cor.2019.05.006.

Lu, X., Madadi, B., Farah, H., Snelder, M., Annema, J.A., van Arem, B., 2019. Scenario-Based Infrastructure Requirements for Automated Drivingn. In: 19th COTA International Conference of Transportation Professionals. American Society of Civil Engineers (ASCE), pp. 5684–5695. https://doi.org/10.1061/9780784482292.489.

Madadi, B., Van Nes, R., Snelder, M., Van Arem, B., Keyvan-Ekbatani, M., 2021. Optimizing road networks for automated vehicles with dedicated links, dedicated lanes, and mixed-traffic subnetworks. J. Adv Transp. 2021, 1–17. https://doi.org/10.1155/2021/8853583.

Madadi, B., Nes, R., Snelder, M., Arem, B., 2020. A bi-level model to optimize road networks for a mixture of manual and automated driving: an evolutionary local search algorithm. Comput. -Aided Civ. Infrastruct. Eng. 35 (1), 80–96. https://doi.org/10.1111/mice.12498.

Madadi, B., van Nes, R., Snelder, M., van Arem, B., 2019. Assessing the travel impacts of subnetworks for automated driving : an exploratory study. Case Stud. Transp. Policy 7, 48–56. https://doi.org/10.1016/j.cstp.2018.11.006.

Magnanti, T.L., Wong, R.T., 1984. Network design and transportation planning: models and algorithms. Transp. Sci. 18 (1), 1–55.

Miandoabchi, E., Daneshzand, F., Zanjirani Farahani, R., Szeto, W.Y., 2015. Time-dependent discrete road network design with both tactical and strategic decisions. J. Oper. Res. Soc. 66 (6), 894–913. https://doi.org/10.1057/jors.2014.55.

Miandoabchi, E., Daneshzand, F., Szeto, W.Y., Zanjirani Farahani, R., 2013. Multi-objective discrete urban road network design. Comput. Oper. Res. 40 (10), 2429–2449. https://doi.org/10.1016/j.cor.2013.03.016.

Miandoabchi, E., Farahani, R.Z., Dullaert, W., Szeto, W.Y., 2012. Hybrid evolutionary metaheuristics for concurrent multi-objective design of urban road and public transit networks. Netw. Spat. Econ. 12 (3), 441–480. https://doi.org/10.1007/s11067-011-9163-x.

Migdalas, A., 1995. Bilevel programming in traffic planning: models, methods and challenge. J Glob Optim 7 (4), 381–405. https://doi.org/10.1007/BF01099649.

Nieuwenhuijsen, J., Correia, G.H.d.A., Milakis, D., van Arem, B., van Daalen, E., 2018. Towards a quantitative method to analyze the long-term innovation diffusion of automated vehicles technology using system dynamics. Transp. Res. Part C: Emerg. Technol. 86, 300–327. https://doi.org/10.1016/j.trc.2017.11.016.

O'Brien, L., Szeto, W.Y., 2007. The Discrete Network Design Problem Over Time. HKIE Transactions 14.

SAE International, 2018. Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles.

Shladover, S.E., 2016. The Truth about "Self-Driving" Cars. Sci. Am. 314, 52–57. https://doi.org/10.1038/scientificamerican0616-52.

Szeto, W.Y., Jaber, X., O'Mahony, M., 2010. Time-dependent discrete network design frameworks considering land use. Comput. -Aided Civ. Infrastruct. Eng. 25, 411–426. https://doi.org/10.1111/j.1467-8667.2010.00654.x.

Tobin, R.L., Friesz, T.L., 1988. Sensitivity analysis for equilibrium network flow. Transp. Sci. 22 (4), 242–250.

Ukkusuri, S.V., Patil, G., 2009. Multi-period transportation network design under demand uncertainty. Transp. Res. Part B: Methodol. 43, 625–642. https://doi.org/10.1016/j.trb.2009.01.004.

Wang, G., Gao, Z., Xu, M., Sun, H., 2014. Models and a relaxation algorithm for continuous network design problem with a tradable credit scheme and equity constraints. Comput. Oper. Res. 41, 252–261. https://doi.org/10.1016/j.cor.2012.11.010.

Wang, J., Peeta, S., He, X., 2019. Multiclass traffic assignment model for mixed traffic flow of human-driven vehicles and connected and autonomous vehicles. Transp. Res. Part B: Methodol. 126, 139–168. https://doi.org/10.1016/j.trb.2019.05.022.

Wang, S., Meng, Q., Yang, H., 2013. Global optimization methods for the discrete network design problem. Transp. Res. Part B: Methodol. 50, 42–60. https://doi.org/10.1016/j.trb.2013.01.006.

Wang, Y.i., Szeto, W.Y., 2017. Multiobjective environmentally sustainable road network design using Pareto optimization: environmentally sustainable road network design using Pareto optimization. Comput.-Aided Civ. Infrastruct. Eng. 32 (11), 964–987. https://doi.org/10.1111/mice.12305.

Wardrop, J., 1952. Some theoretical aspects of road traffic research. Proceedings of the Institution of Civil Engineers.

Wu, J.H., Florian, M., He, S., 2006. An algorithm for multi-class network equilibrium problem in PCE of trucks: application to the SCAG travel demand model. Transportmetrica 2 (1), 1–9. https://doi.org/10.1080/18128600608685656.

Xu, C., Ding, Z., Wang, C., Li, Z., 2019. Statistical analysis of the patterns and characteristics of connected and autonomous vehicle involved crashes. J. Saf. Res. 71, 41–47. https://doi.org/10.1016/j.jsr.2019.09.001.

Yang, H., 1997. Sensitivity analysis for the elastic-demand network equilibrium problem with applications. Transp. Res. Part B: Methodol. 31 (1), 55–70. https://doi.org/10.1016/S0191-2615(96)00015-X.

Yang, H., Bell, H., 1998. Models and algorithms for road network design: a review and some new developments. Transp. Rev. 18 (3), 257–278. https://doi.org/10.1080/01441649808717016.

Yang, H., Meng, Q., 2001. Modeling user adoption of advanced traveler information systems: dynamic evolution and stationary equilibrium. Transp. Res. Part A 35, 895–912.

Ye, Y., Wang, H., 2018. Optimal design of transportation networks with automated vehicle links and congestion pricing. J. Adv. Transp. 2018, 1–12. https://doi.org/10.1155/2018/3435720.

Zhou, Z., Chen, A., Wong, S.C., 2009. Alternative formulations of a combined trip generation, trip distribution, modal split, and trip assignment model. Eur. J. Oper. Res. 198 (1), 129–138. https://doi.org/10.1016/j.ejor.2008.07.041.